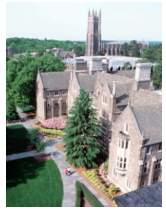
Computing and Analysis Model at CMS

D. Bonacorsi *(INFN-CNAF Tier-1, Bologna, Italy)*

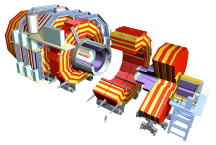
On behalf of the CMS experiment

Duke
UNIVERSITY

HCP06
Hadron Collider Physics Symposium
22-26.May.06 - Duke University, Durham, North Carolina (US)

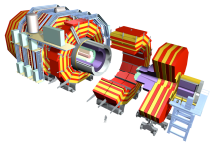


Outline



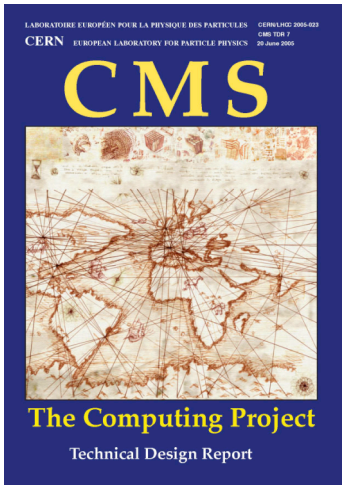
- The CMS distributed computing system
  - from guiding principles to architectural design
- Services, actors and workflows in CMS computing
  - Data Management (DM) and Workload Management (WM)
- The realization of the CMS Computing Model in a Grid-enabled world
  - Implementation of production-level systems on the Grid
    - Data Distribution, MonteCarlo (MC) production, Data Analysis
  - Computing challenges
- Plans towards the LHC data taking

CMS computing model



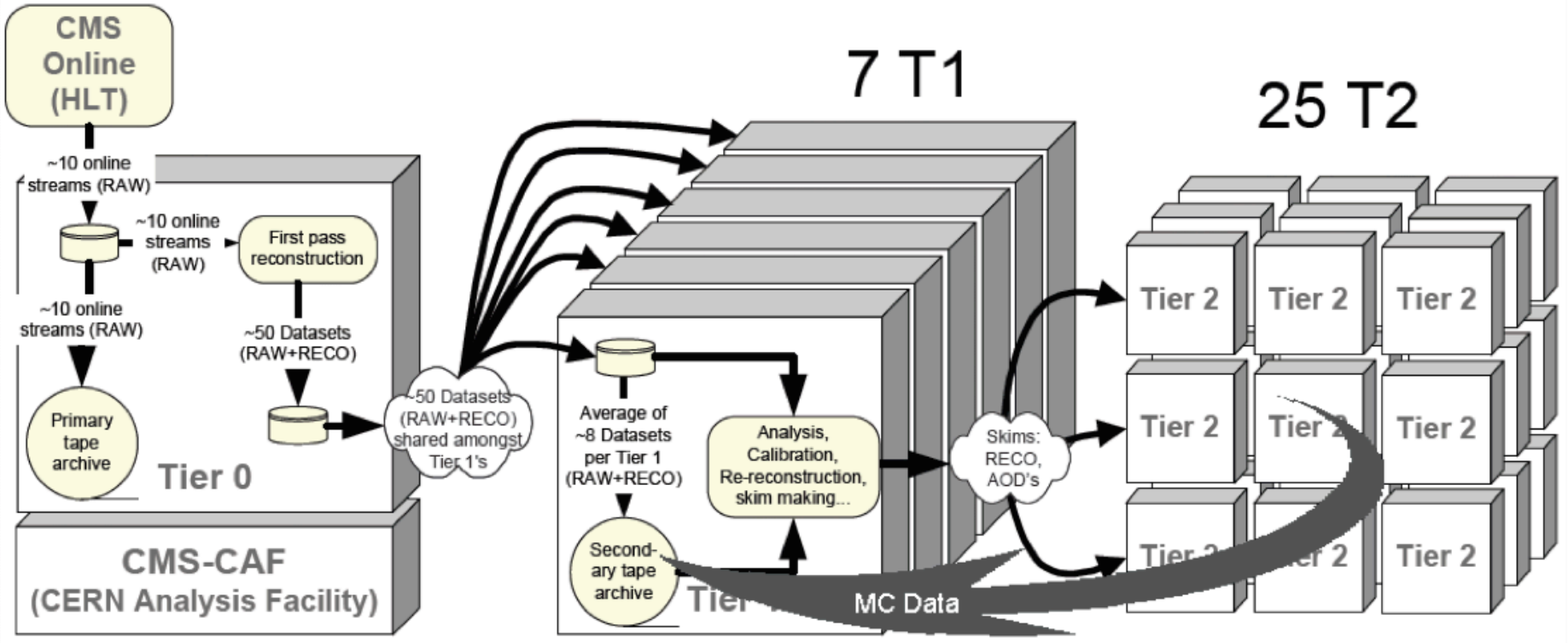
- The CMS computing system relies on a *distributed infrastructure* of Grid resources, services and toolkits
  - distributed system to cope with computing requirements for storage, processing and analysis of data provided by LHC experiments
  - building blocks provided by Worldwide LHC Computing Grid [WLCG]
    - CMS builds application layers able to interface with few - at most - different Grid flavors (LCG-2, Grid-3, EGEE, NorduGrid, OSG)

- CMS computing model document (CERN-LHCC-2004-035)
- CMS C-TDR released (CERN-LHCC-2005-023)
  - in preparation for the first year of LHC running (2008)
    - not "blueprint", but "baseline" targets (+ devel. strategies)
  - hierarchy of computing tiers using WLCG tools
    - focus on Tiers role, functionality and responsibility



Tiered architecture

- T0:
  - Accepts data from DAQ
  - Prompt reconstruction
  - Data archive and distribution to T1's
- CAF (CERN Analysis Facility for CMS):
  - Access to full raw dataset
  - Focused on latency-critical detector trigger calibration and analysis activities
  - Provide some CMS central services (e.g. store conditions and calibrations)



- 7 T1 centers and 25 T2 centers (see next slide)

T1/T2 roles and computing capacities

CMS T1 functions

- Scheduled data-reprocessing and data-intensive analysis tasks:
  - later-pass reco, AOD extraction, skimming, …
- Data archiving (real+MC):
  - custody of raw+reco & subsequently produced data
- Disk storage management:
  - fast cache to MSS, buffer for data transfer, …
- Data distribution:
  - data serving to Tier-2's for analysis
- Analysis:
  - proficient data access via CMS+WLCG services

CMS T2 functions

- User data analysis
- Fast and detailed MC event prod
- Import skimmed datasets from T1s and export MC data
- Data processing for calib/align tasks and detector studies

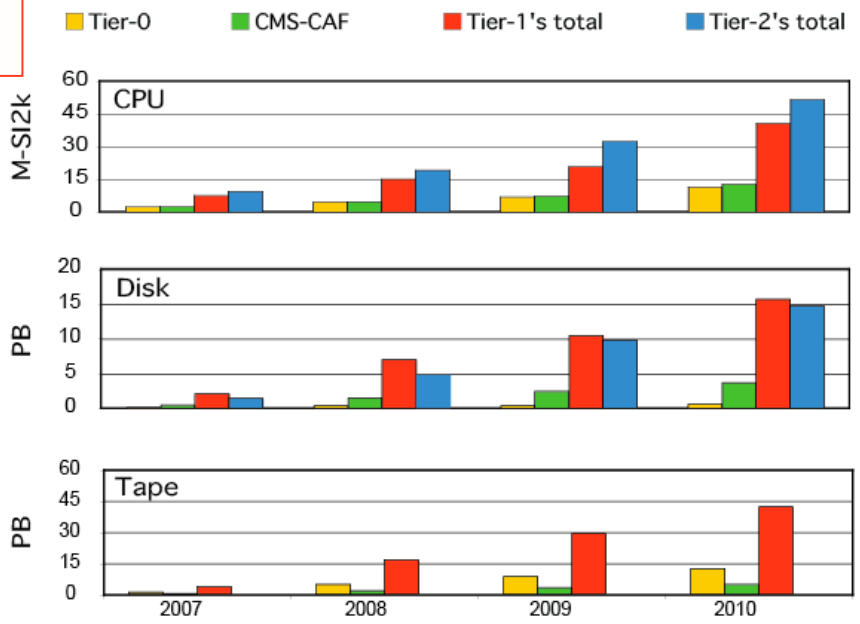
CMS T1 resources (nominal for average T1 in 2008):

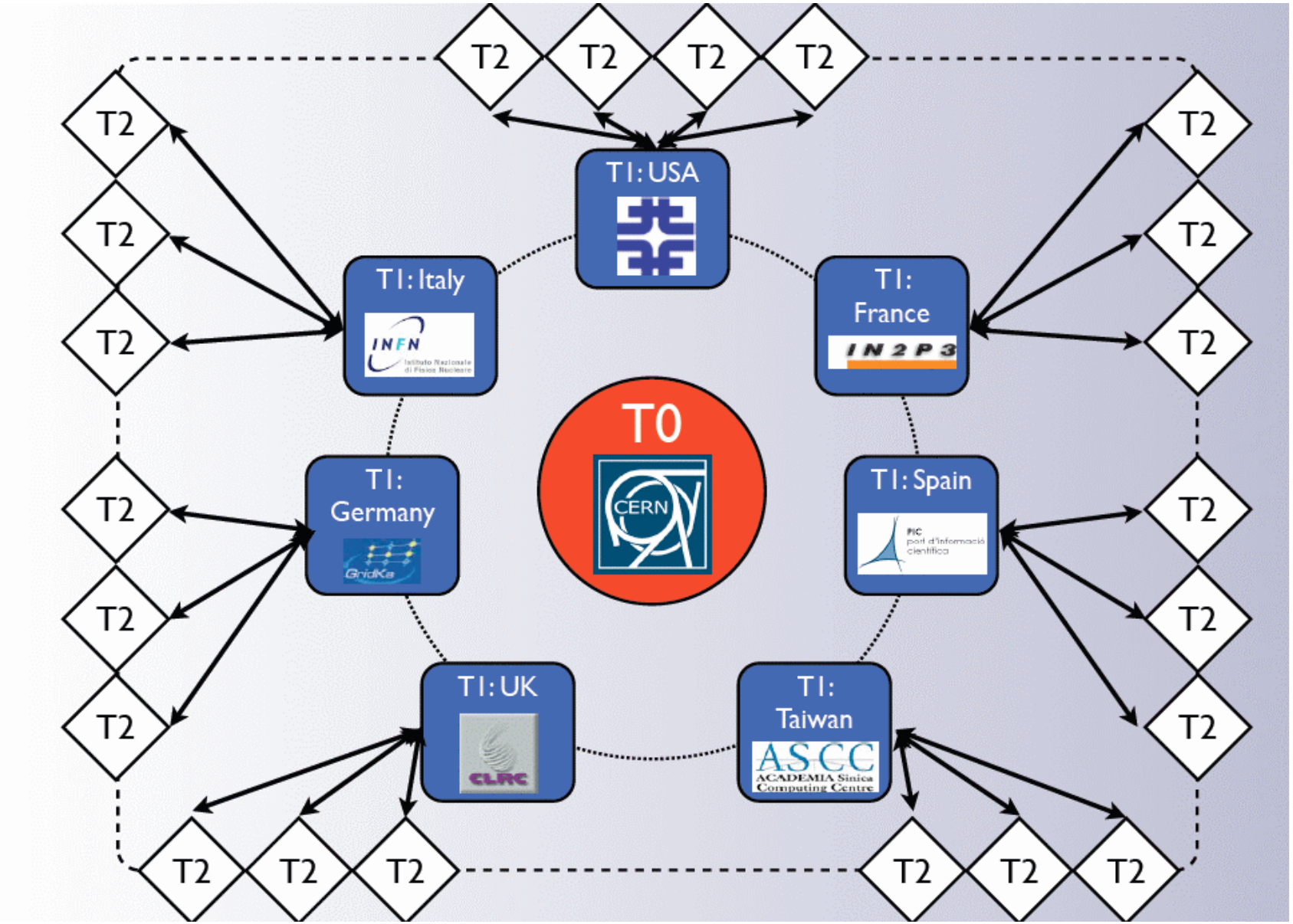
NB:1/7

- ✓ WAN: transfer capacity ~10 Gb/s
- ✓ CPU: 2.5 M-SI2k (scheduled reprocessing : analysis = 2 : 1)
- ✓ Disk: 0.8 PB (~85% for analysis data serving)
- ✓ MSS: 2.8 PB (losses ~tens of GB per PB stored)

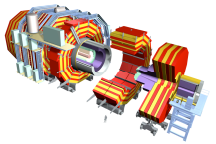
CMS T2 resources (nominal for average T2 in 2008):

- ✓ WAN: 1 Gb/s (at least)
- ✓ CPU: 900 k-SI2k
- ✓ Disk: 200 TB




T2
T2
T2
T2
T1: USA
T2
T2
T2
T1: Italy
INFN
T1: France
IN2P3
T2
T2
T0
CERN
T2
T2
T1: Germany
GridKa
T1: Spain
PIC
T2
T2
T2
T1: UK
CLRC
T1: Taiwan
ASCC
ACADEMIA Sinica Computing Centre
T2
T2
T2
T2
T2
T2

Data-driven baseline



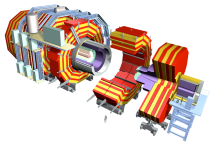
Technical baseline principles

- Baseline system with minimal functionality for first physics
  - 'Keep it simple!'
  - Use Grid services as much as possible + also CMS-specific services
  - Optimize for the common case
    - for read access (most data is write-once, read-many)
    - for organized bulk processing, but without limiting single user
  - Decouple parts of the system
    - Minimise job dependencies + Site-local information remain site-local

- T0-T1s activities driven by **data placement** in the CMS baseline model
  - Data is partitioned by the exp as a whole, do not move around in response to job submission, all data is placed at a site through explicit CMS policy
  - Tier-0 and Tier-1 are resources for the whole experiment
  - Leads to very 'structured' usage of Tier-0 and Tier-1
    - T0/T1s are CMS experiment resources and their activities and functionality are largely predictable since nearly entirely specified
      - i.e. organized mass processing and custodial storage

- 'unpredictable' computing essentially restricted to data analysis at T2s
  - T2s are the place where more flexible, user driven activities can occur
  - Very significant computing resources and good data access are needed

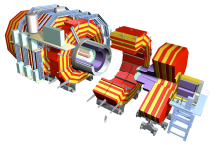
Guiding principles



- Prioritization will be important
  - In 2007/8, computing system efficiency may not be 100%
  - cope with potential reconstruction backlogs without delaying critical data
  - Reserve possibility of 'prompt calibration' using low-latency data
  - Also important after first reconstruction, and throughout the system
    - e.g. for data distribution, 'prompt' analysis
- Streaming
  - Classifying events early allows prioritization and data access optimization
    - e.g. 'express stream' of hot / calibration events
  - Propose O(10) 'online streams', O(2PB)/yr raw data split into O(50) (40 TB) 'primary' trigger-determined datasets
- Baseline principles for 2008
  - Fast reconstruction code (i.e. 'reconstruct often')
  - Streamed primary datasets
  - Efficient workflow and bookkeeping systems
  - Distribution of RAW and RECO data together
  - Compact data format AOD (multiple distributed copies)

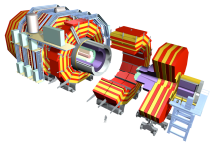
Next: data organization + feed the model with numbers… (see next slides)

Data organization



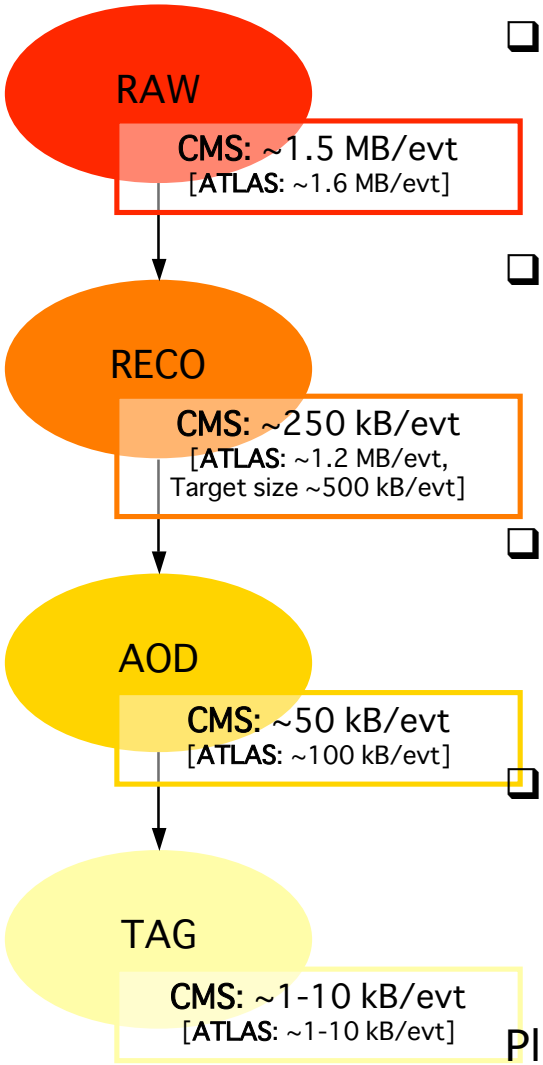
- CMS expects to produce large amounts of data (events)
  - O(PB)/year
- Event data are in **files**
  - average file size is kept reasonably large (≥ GB)
    - avoid scaling issues with storage systems and catalogues when dealing with too many small files (+ foresee file merging)
  - O(10^6) files/year
- Files are grouped in **fileblocks**
  - group files in blocks (1-10 TB) for bulk data management reasons
    - exist as a result of either MC production or data movement
  - 10^3 Fileblocks/year
- Fileblocks are grouped in **datasets**
  - Datasets are large (100 TB) or small (0.1TB)
    - Dataset definition is physics-driven (size as well)

Data types



➢ Data tiers/volumes for 2008 as input parameters for the model*

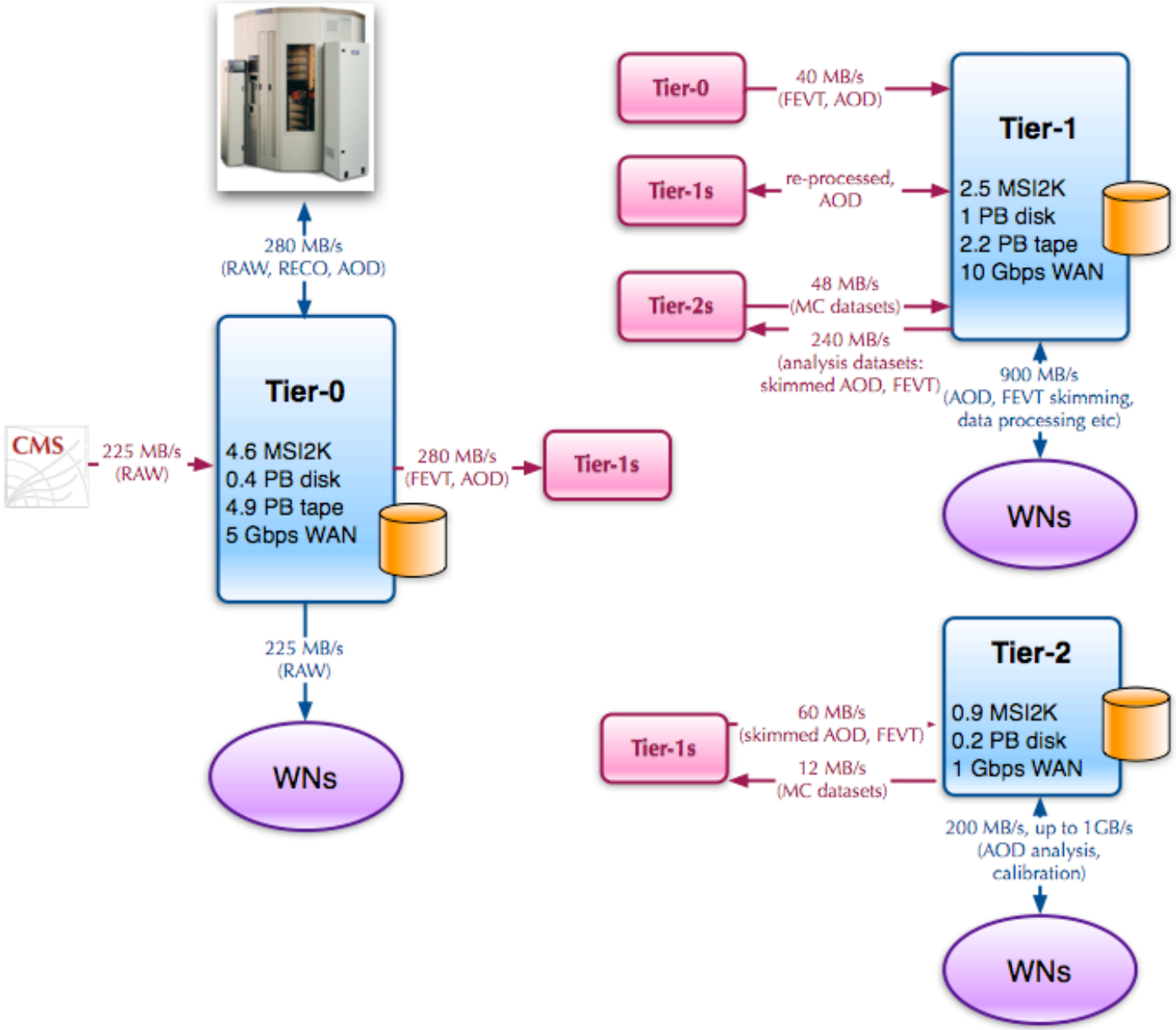
[*] safety factors included (poor understanding of the detector, compression, …)



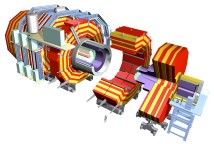
- **RAW**
  - Triggered evts recorded by DAQ
  - ~1.5 MB/evt @ ~150 Hz; ~ 4.5 PB/yr
    - 2 copies: 1 at T0 (archive all, serve some) and 1 spread over T1s (archive all, serve all)
- **RECO**
  - Reconstructed objects with their associated hits
    - Detailed output of the detector reco: track candidates, hits, cells for calib
  - ~250 kB/evt; ~ 2.1 PB/yr (incl. reprocessing)
    - 1 copy spread over T1s (together with associated RAW)
- **AOD** (Analysis Object Data)
  - Main analysis format: objects + minimal hit info
    - Summary of the reco evt for common analyses: particles id, jets, …
  - ~50 kB/evt; ~ 2.6 PB/yr
    - Whole set copied to each T1, large fraction copied to T2
- **TAG**
  - Fast selection info
    - Relevant info for fast evt selection in AOD
  - ~1-10 kB/ev

Plus MC in ~ N:1 ratio with data

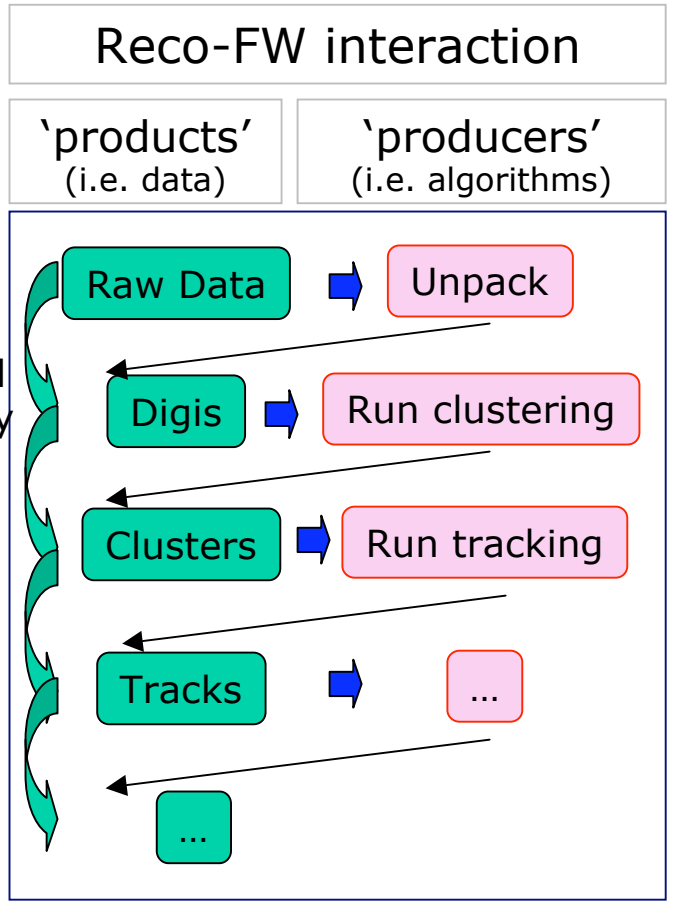
CMS data flows



CMS Framework and Software

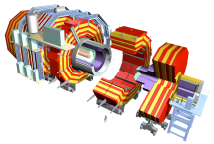


- OO approach to develop framework and software
  - common and basic principles:
    - Abstract interfaces (C++)
    - Clear separation between data/algos
- Input from CMS DC04: critical issues identified in CMS software design
  - Not simple to perform analysis using ROOT (+ external libs) or interactive analysis, no explicit scheduling (only on-demand reco)
- CMS decided to reengineer its software
  - Main goal: provide a reco sw with high modularity, predefined scheduling and allowing the direct use of ROOT in the FW, whose structure is completely changed
- Now CMS had 2 lines of sw development:
  - Old sw was still used to provide results for the P-TDR II
  - New sw (CMSSW) under development



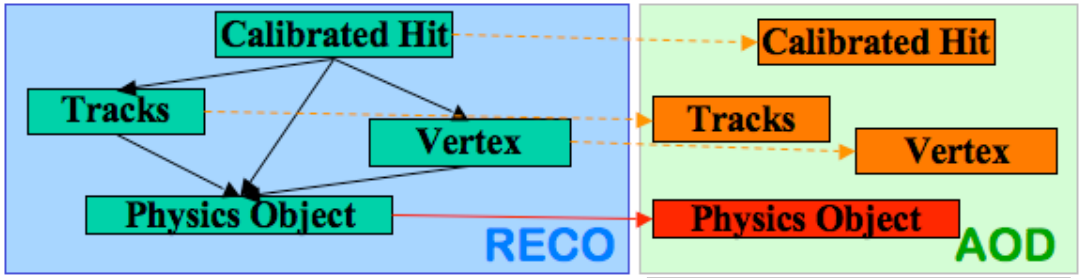
ATLAS: persistent (in file) vs. transient (in mem) data
More flexibility
CMS: use the same for both
Better performances

Analysis Model, basic types, EDM requirements



- CMS is starting from concrete experiences to try to build a usable, effective and user-friendly AM
  - CMS started from DC04 and P-TDR experiences…
    - understand possible use-cases and main requirements for the AM
  - … and will go on through CSA06 (and exp parts of WLCG SCs)

Content (⇒ size) of RECO/AOD is currently evolving due to increasing knowledge of what's actually needed for analysis



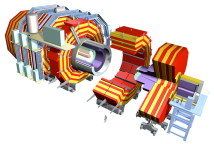
E.g. AOD needs balancing among two main constraints:

① AOD size constraint (from the CMS computing model)

② "For my analysis I am fine with current AOD content in X% of cases": do maximize X

- CMS analysis basic types put requirements on the EDM:
  - *AOD*:
    - a condensed format, a subset of RECO (i.e. tracks in AOD, tracks+hits in RECO) of physics objects in order to use it in the final analysis
    - main idea is capability to read data directly using ROOT w or w/o loading shared libs
  - *Particles Candidates*:
    - to be built on top of either RECO or AOD;
  - *User Data*:
    - user should be allowed to add his own data to the Event, either persistent quantities that require large processing time or ntuple-like formats for interactive access

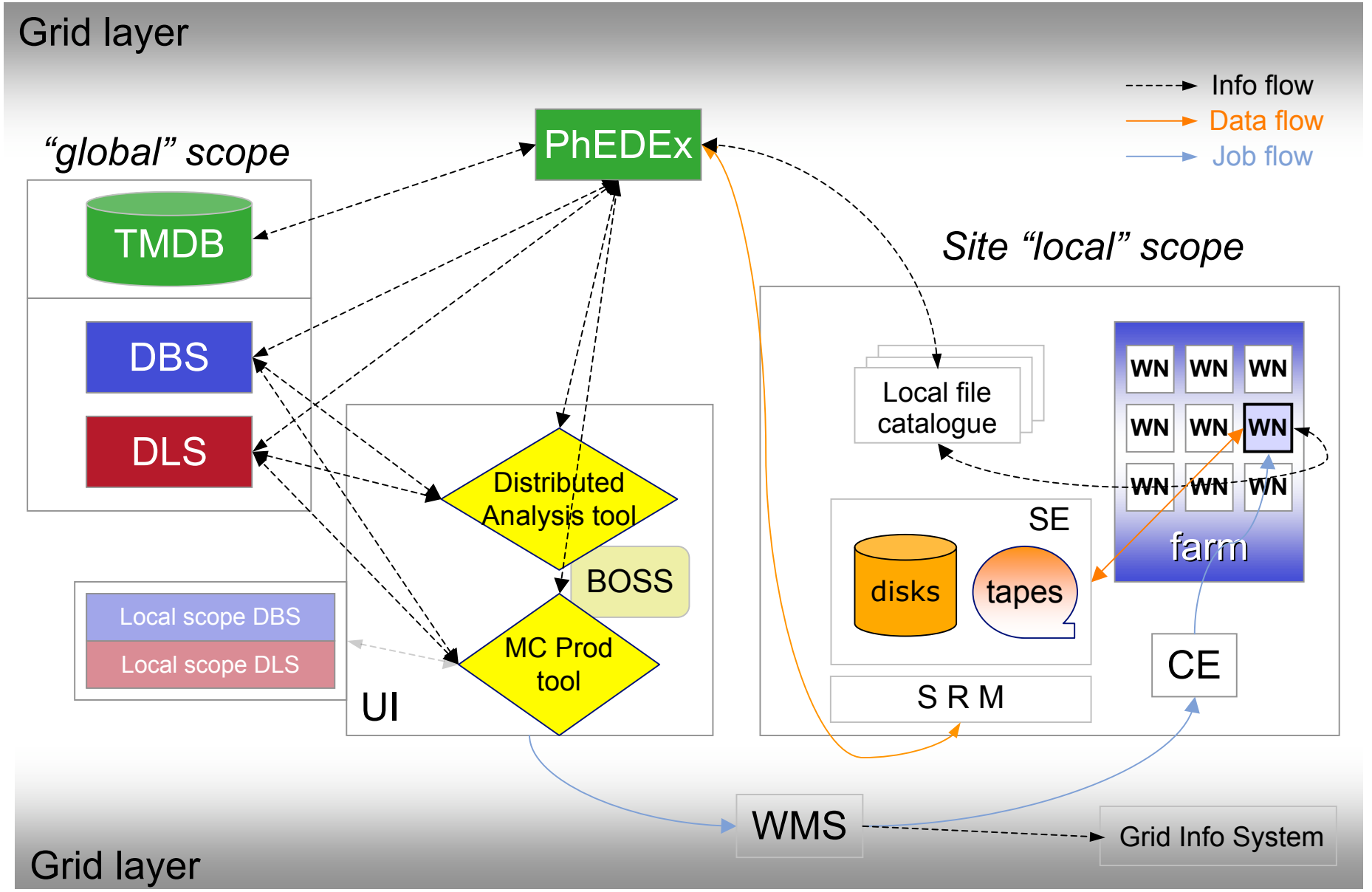
Work-in-progress on WMS&DMS



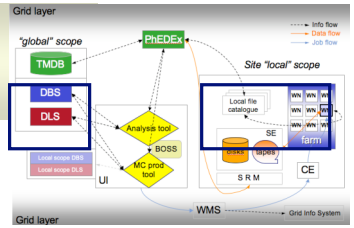
- Migration from current DMS to new DMS
  - Provide new tools to discover, access and transfer event data in a distributed computing environment
    - Track and replicate data with a granularity of file blocks
    - Reduce load on catalogues
  - DBS (Dataset Bookkeeping system)
    - DBS provides the means to define, discover and use CMS event data
  - DLS (Dataset Location Service)
    - DLS provides the means to locate replicas of data in the distributed system
  - local file catalogue solutions
  - PhEDEx integration with gLite FTS
    - PhEDEx takes care of reliable, scalable CMS dataset replication (and more…)
    - FTS takes care of reliable point-to-point transfers of files
- New DMS to be exercised with new MC production system
  - integrate with new Event Data Model and new DMS

[the migration was not disruptive: old DMS was kept for PTDR analyses …]

Data processing workflow



DBS / DLS / local file catalogue



Data definition:
- dataset specification (content and associated metadata)
- track data provenance

Data discovery:
- which data exist
- dataset organization (in term of fileblocks/files)
- site independent information

"What data exists?"

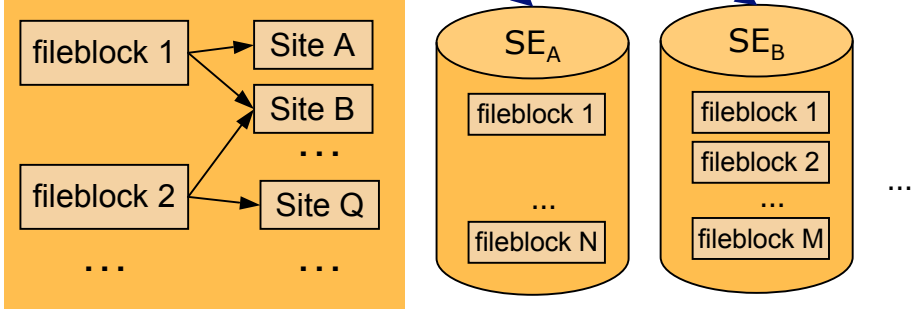
DBS

Interaction with DBS:
- Distributed analysis tool
- MC Production system
- PhEDEx for injection
- User query

"Where is data located?"

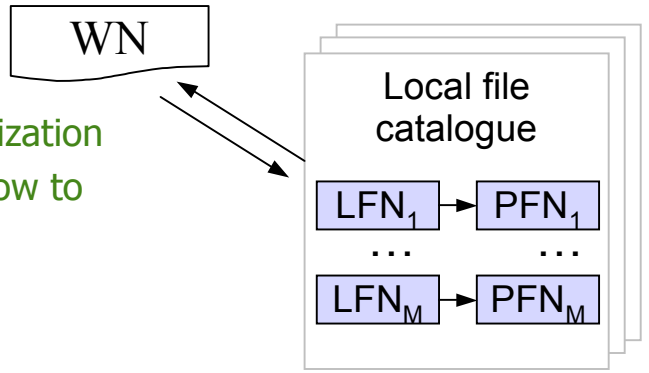
DLS

Integration with DLS:
- Insert file-blocks produced at a site
- Insert file-blocks upon data replication
- Query to locate file-blocks (e.g. analysis tool)

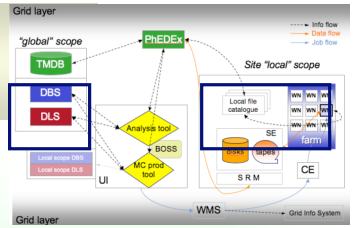


DLS maps fileblocks to SEs where they are located

- Need a catalogue + a site local discovery mechanism
  - discover at runtime on WN the site-dependent data organization
  - local file catalogues provide site local information about how to access any given file (aka "**LFN-to-PFN mapping**")
    - CMS baseline solution is to use a trivial file catalogue
    - High-rate large-scale performances required



DBS / DLS / local file catalogue



Data definition:
- dataset specification
  (content and associated metadata)
- track data provenance

Data discovery:
- which data exist
- dataset organization
  (in term of fileblocks/files)
- site independent information

"What data exists?"

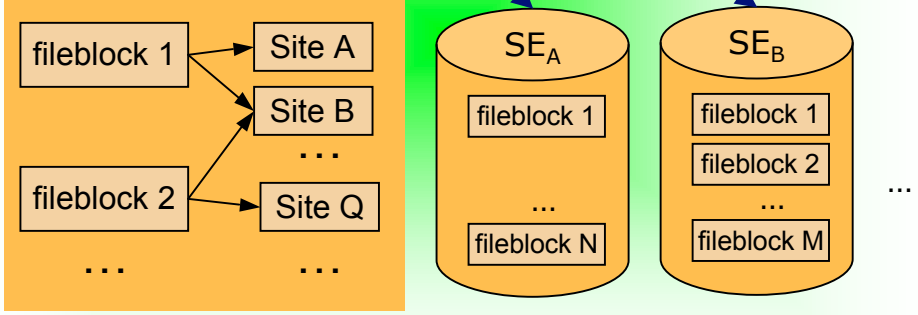
DBS

Interaction with DBS:
- Distributed analysis tool
- MC Production system
- PhEDEx for injection
- User query

"Where is data located?"

DLS

Integration with DLS:
- Insert file-blocks produced at a site
- Insert file-blocks upon data replication
- Query to locate file-blocks
  (e.g. analysis tool)

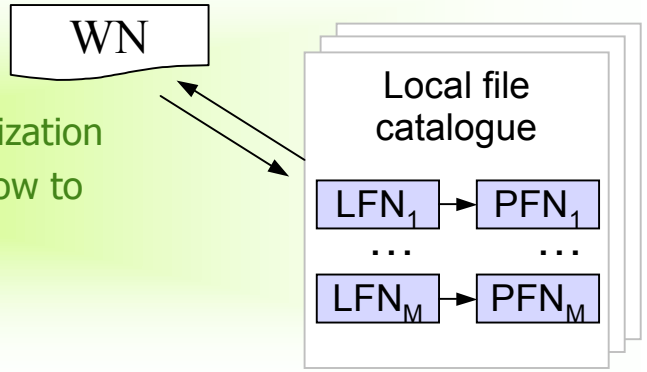


DLS maps fileblocks to SEs where they are located

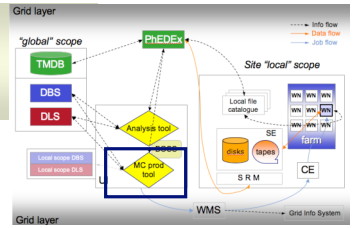
site independent | **site dependent**

site dependent job configuration

- Need a catalogue + a site local discovery mechanism
  - discover at runtime on WN the site-dependent data organization
  - local file catalogues provide site local information about how to access any given file (aka "**LFN-to-PFN mapping**")
    - CMS baseline solution is to use a trivial file catalogue
    - High-rate large-scale performances required



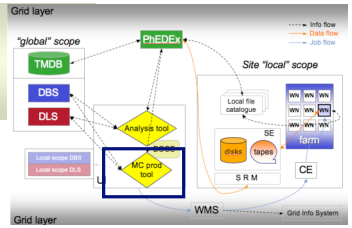
New MC Production system



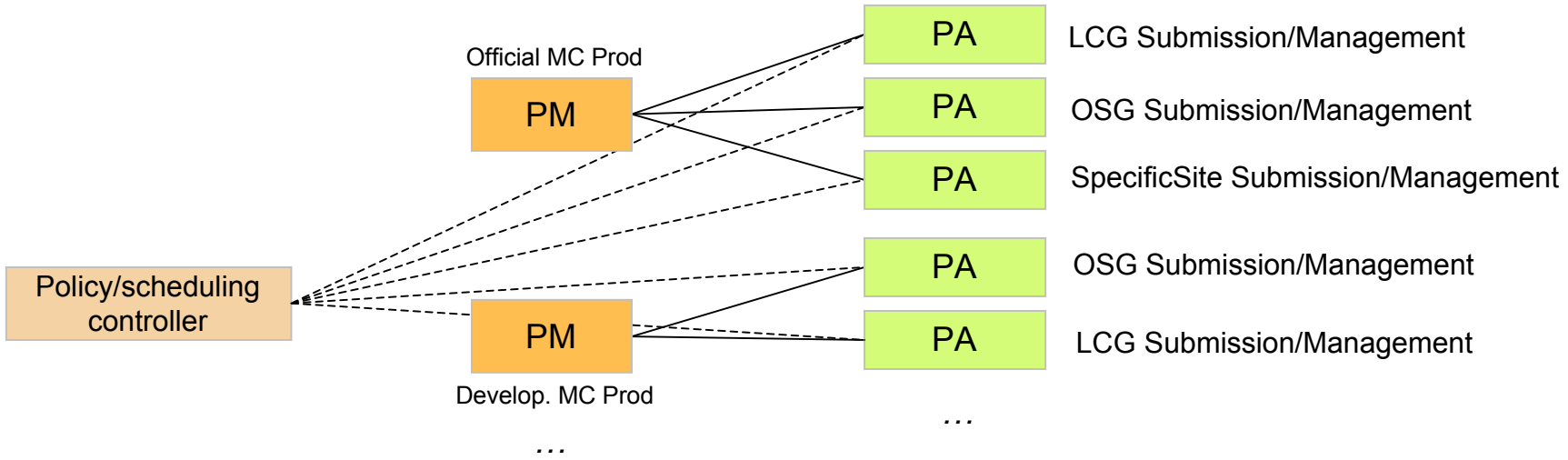
- Large experience gained running McRunjob
  - Designed for local farm and ported to Grid in 2005
    - ~ 10M events/month (4x10K jobs), ~ 150M events in total
    - ~ 20% in OSG and 15% in LCG, the rest on local farms of big sites
      - although mostly production on the Grid in the last months

- New MC production system being developed
  - Overcome current inefficiencies + introduce new capabilities
    - less man-power consuming, better handling of Grid-sites unreliability, better use of resources, automatic retrials, better error report/handling
  - integrate with **new Event Data Model** and **new DMS**
    - Bringing up a DBS system capable of being used for MC production with the new EDM + data merging, fileblock management
    - job chaining, e.g. generation→simulation→digitization→reconstruction
    - Orchestrate the interactions with local scope DBS/DLS and data placement system

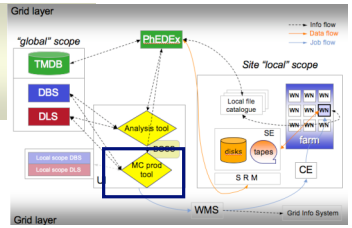
MC Production system: architecture



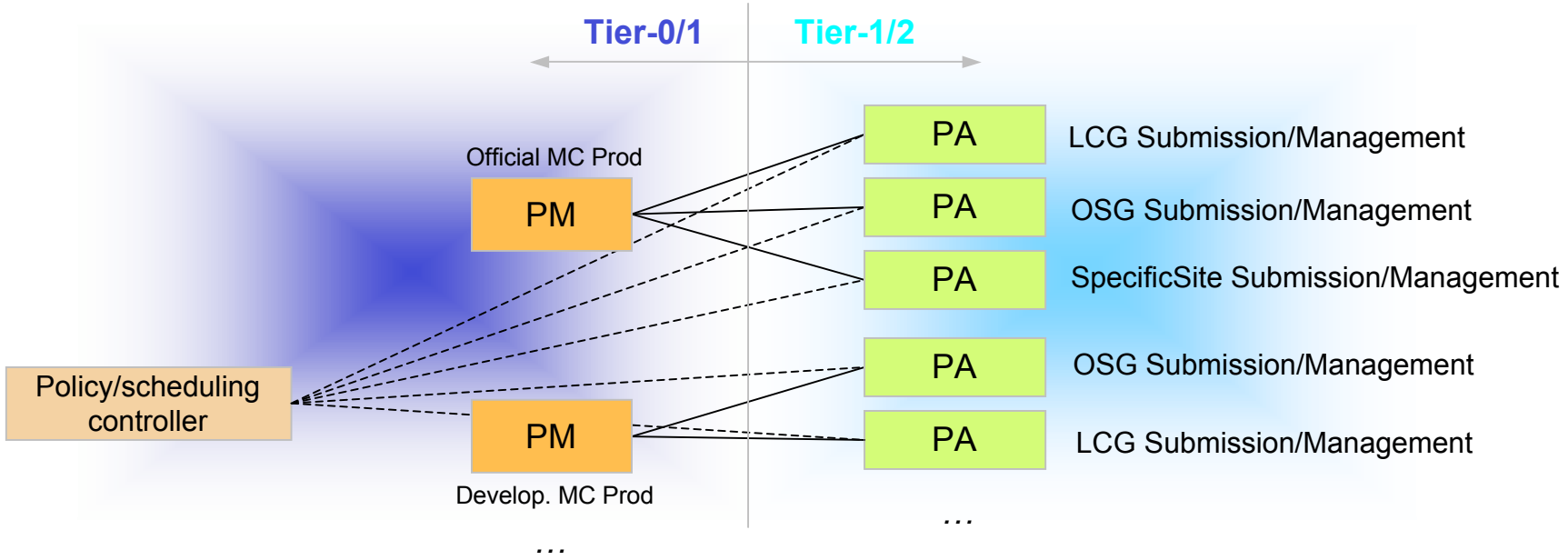
- More flexible and automated architecture
  - **ProdManager** (*PM*) (+ the policy piece)
    - manage the assignment of requests to 1+ *ProdAgents* and tracks the global completion of the task
  - **ProdAgent** (*PA*)
    - Job creation, submission and tracking, management of merges, failures, resubmissions, …
      - It works with a set of resources (e.g. a Grid, a Site)



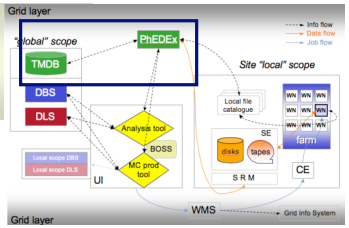
MC Production system: architecture



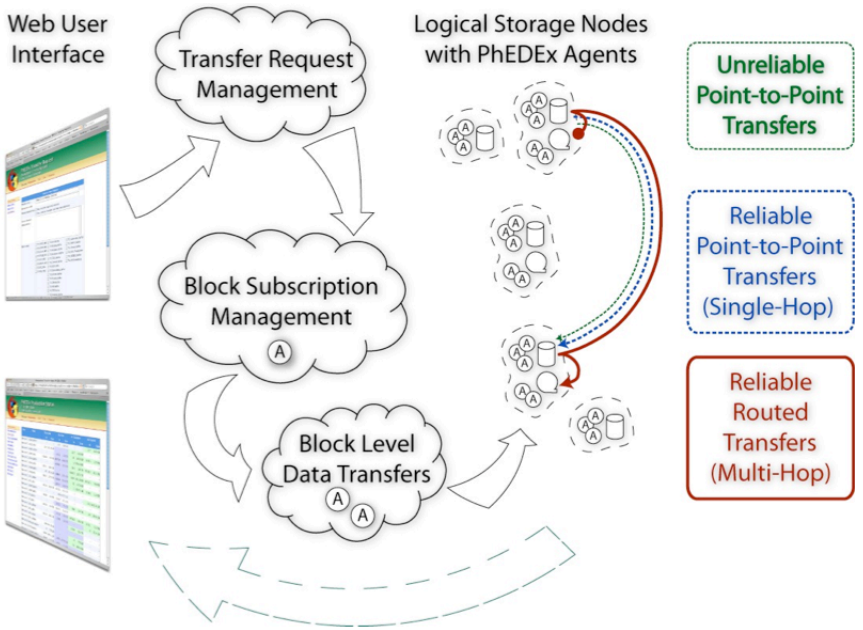
- More flexible and automated architecture
  - **ProdManager** (*PM*) (+ the policy piece)
    - manage the assignment of requests to 1+ *ProdAgents* and tracks the global completion of the task
  - **ProdAgent** (*PA*)
    - Job creation, submission and tracking, management of merges, failures, resubmissions, …
      - It works with a set of resources (e.g. a Grid, a Site)

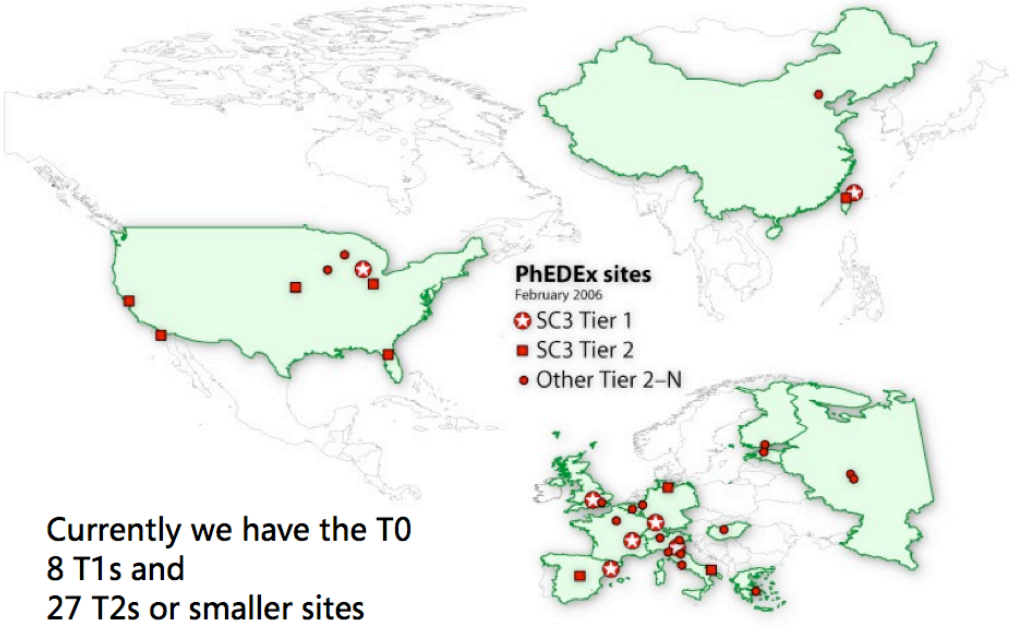


Data transfer and placement system

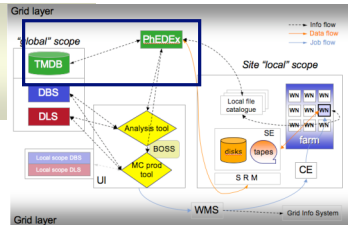


- Physics Experiment Data Export (PhEDEx)
  - large scale reliable dataset/fileblock replication
    - multi-hop routing following a transfer topology (T0 → T1's ↔ T2's), data pre-stage from tape, monitoring, bookkeeping, priorities and policy, etc
  - In production since two years
    - Managing transfers of several TB/day
      - ~150 TB known to PhEDEx, ~350 TB total replicated

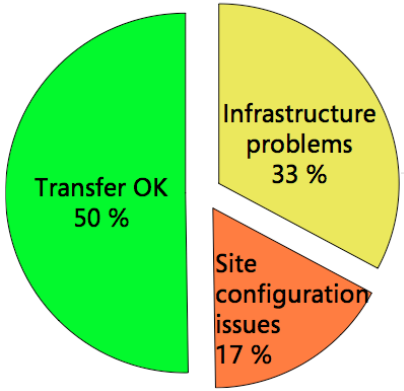




PhEDEx reliability and performances



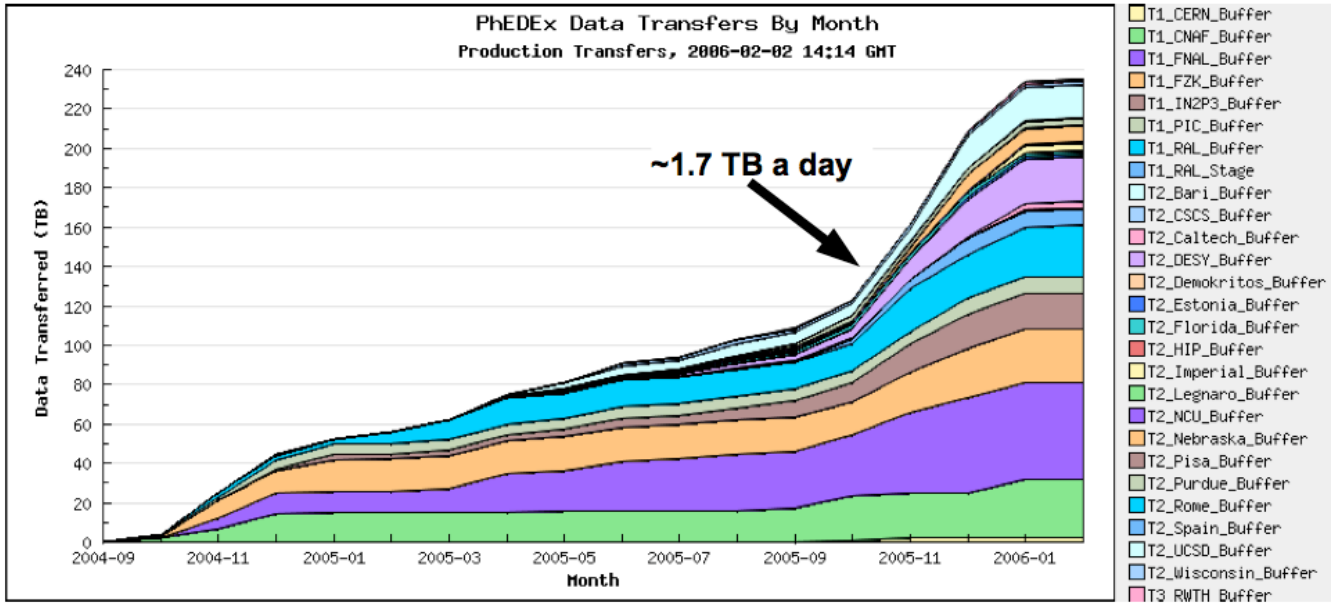
Case study: transfer level



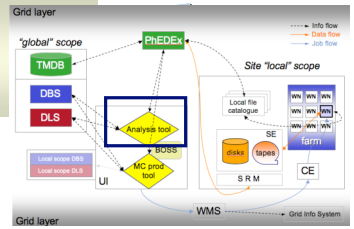
- High failure rate on new SRM/storage infrastructure
- 50% of the transfers successful on the first try
- Main problems
  - Configurations changed or wrong at sites
  - Problems related to network or storage infrastructure

- All failures recovered, eventually
- Files retransferred
- No data lost :-)
- Recovery fully automatic
  - Absolute must: in 2007 CMS will transfer ~2-10k files per day
  - Manual recovery infeasible: 1 ‰ permanent error rate ≃ 2 hrs daily maintenance

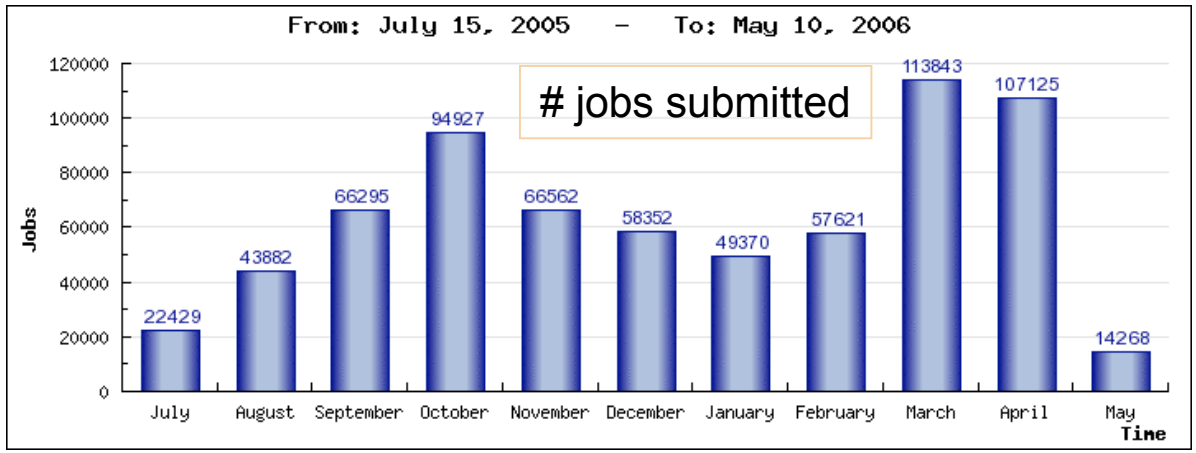
Case study: after PhEDEx failure recovery



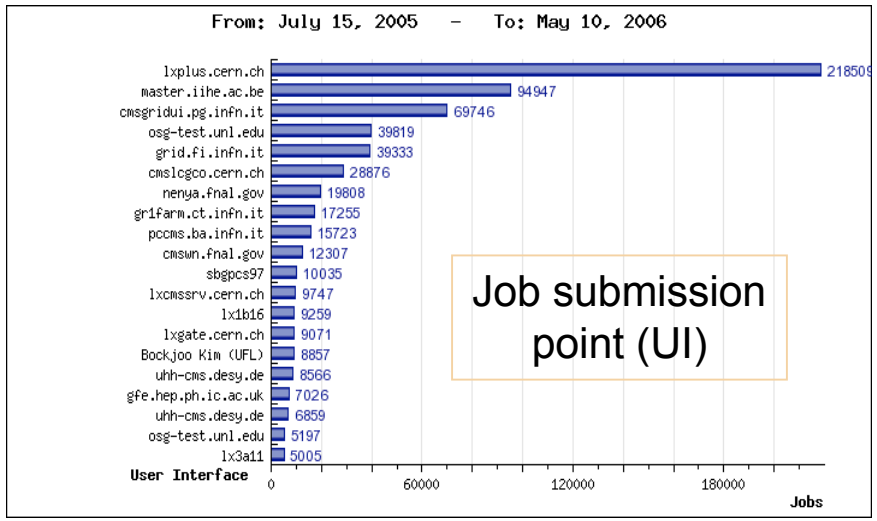
CMS distributed analysis on Grid

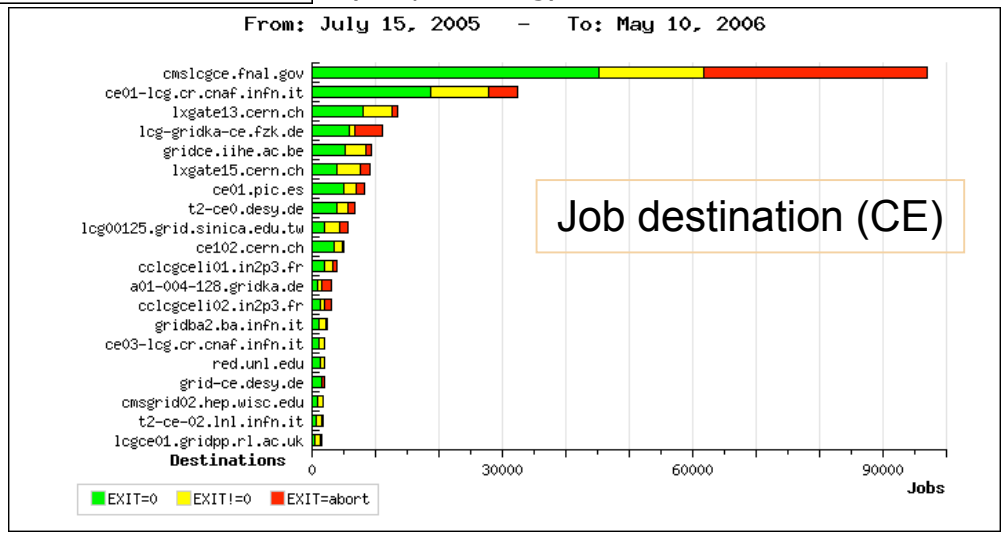


- CMS Remote Analysis Builder (CRAB)
  - Tool for job preparation, submission and monitoring
  - ~ 100K analysis jobs/month (peaks at ~10k/day)

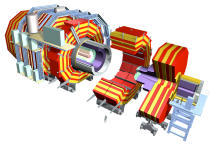


Job destination pattern shows a first example of load balancing depending on data availability (i.e. publishing) at Tiers





Experience from Computing Challenges

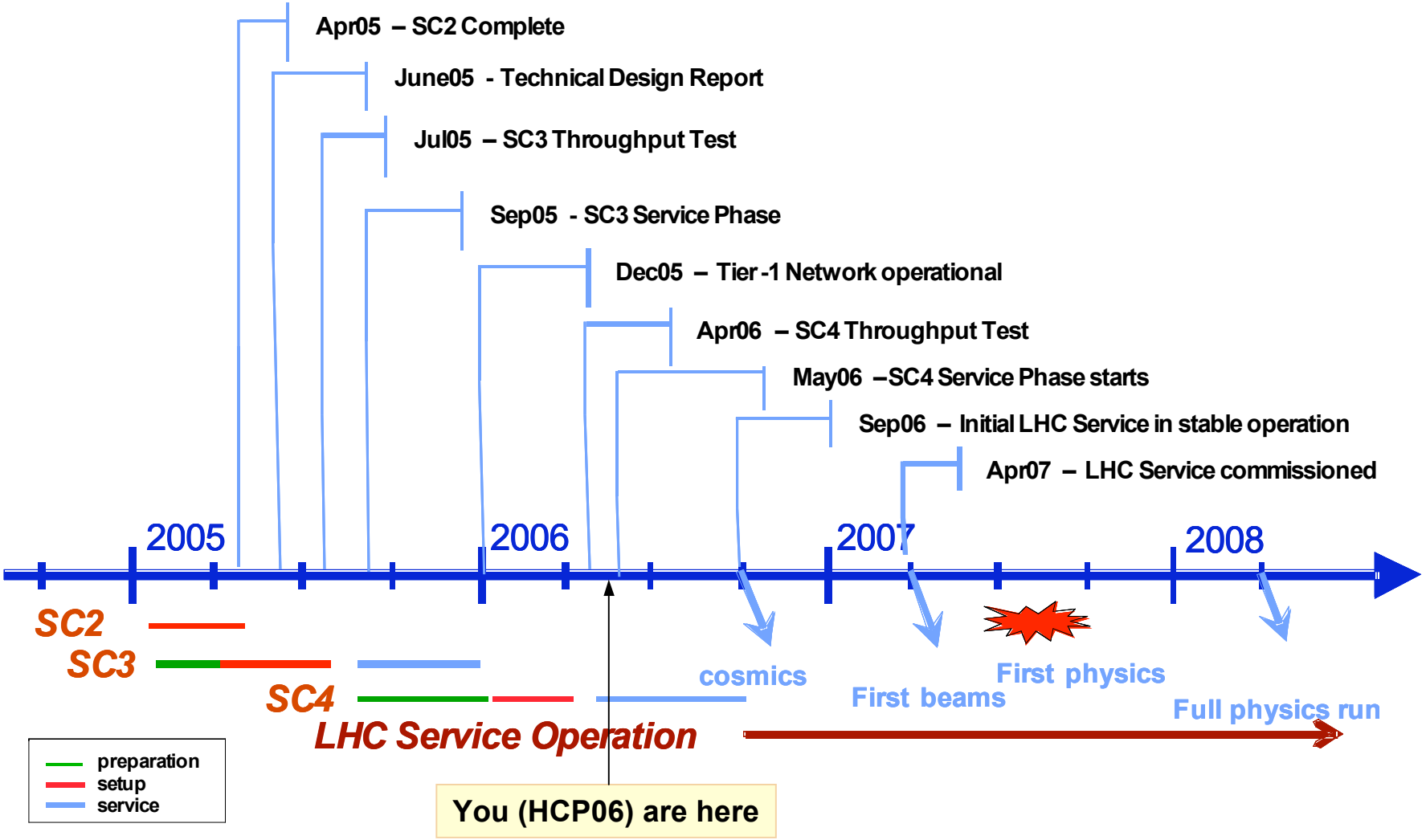


- CMS computing system realization is an iterative process
  - Grid resources/services and CMS solutions for WMS/DMS are tested in scheduled "challenges" of increasing scale and complexity

- Some are indeed CMS-specific…
  - CMS Data Challenge 2004
    - Tier-0 reco @ 25 Hz and data distribution to Tier-1 centers for real-time analysis using Grid interfaces
      - Put in place CMS data transfer and placement system (PhEDEx), first large scale test of Grid WMS (real-time analysis), problems identified: all addressed.
- … some are WLCG-wide
  - WLCG Service Challenges
    - a mechanism by which the readiness of the overall LHC computing infrastructure to meet the exps' requirements is measured and if(/where) necessary corrected
    - understand what it takes to run a **real and wide set of Grid services**
      - Tiers community effort, to trigger resources deployment, drive activity planning and encourage distributed know-how based on realistic use patterns, ramp-up essential grid services to target levels of reliability, availability, scalability, end-to-end performance

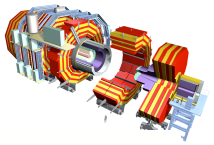
- A long and hard path, done in several steps…
  - Service Challenge 1 and 2 → Focus on T0-T1 infrastructure and services
  - Service Challenge 3 and 4 → Bringing T2s into loop and address exps use-cases

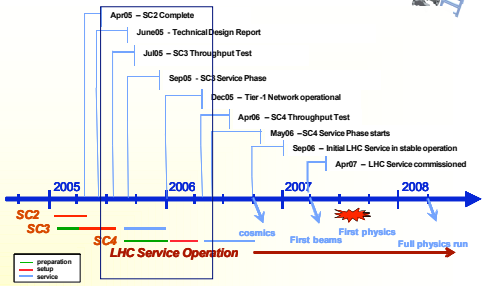
WLCG SC schedule



[figures: courtesy of J.Shiers and WLCG]

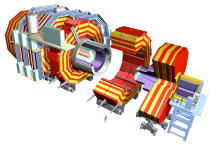
WLCG Service Challenge 3 (SC3)





- Computing integration test exercising the bulk data processing part of the computing model of LHC experiments under realistic conditions
  - Test end-to-end systems of both exp-specific and Grid services
- The first SC with exps-oriented objectives
  - when: Jul 05 - Dec 05 (+ Jan 06)
  - who: T0, all T1's, small nb of T2's
- Set-up ("throughput") phase (Jul 05)
  - Network-to-disk target: 150 MB/s/T1 and 1 GB/s out of CERN
  - Network-to-tape target: 60 MB/s/T1 and 400 MB/s out of CERN
- "Service" phase (Sep-Dec 05)
  - Stable operation during which exps are committed to carry out tests of their sw chains and computing models
  - Includes additional sw components, including a grid WMS, Grid catalogue, mass storage mgmt services and a file transfer service
- Re-run of the throughput phase (Jan 06)

CMS in SC3



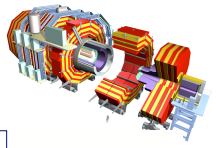
- CMS focused on validation of data storage, transfer and data serving infrastructure plus required workload components for job submission
  - CERN + all 7 CMS T1's + 13 CMS T2's participated
  - A lot of efforts in the service phase

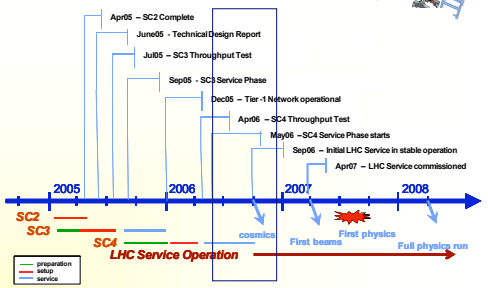
- Results:
  - Data distribution T0 → T1's → T2's
    - Throughput phase: 280 TB, aggregate 200 MB/s sustained for days
    - Service phase: 290 TB, 10-20 MB/s to each T1 on avg over a month
  - Automatic data publishing, validation, analysis at T1's and T2's
    - 70K jobs run. 90% LCG efficiency. Only 60% CMS efficiency
    - Up to 200 MB/s read data throughput from disk to CPU

- Lot of effort spent on debugging and integration
  - Two many underlying Grid and CMS services not sufficiently well prepared to test in a challenge environment. Sites had not verified functionalities

The primary issue for CMS is demostrating that the challenge performance nbs can translate into ***stable*** experiment data transfers
- stabilizing of storage services was a direct benefit of throughput challenges
- capitalizing the service improvements is crucial (see e.g. PhEDEx/FTS integration)

WLCG Service Challenge 4 (SC4)





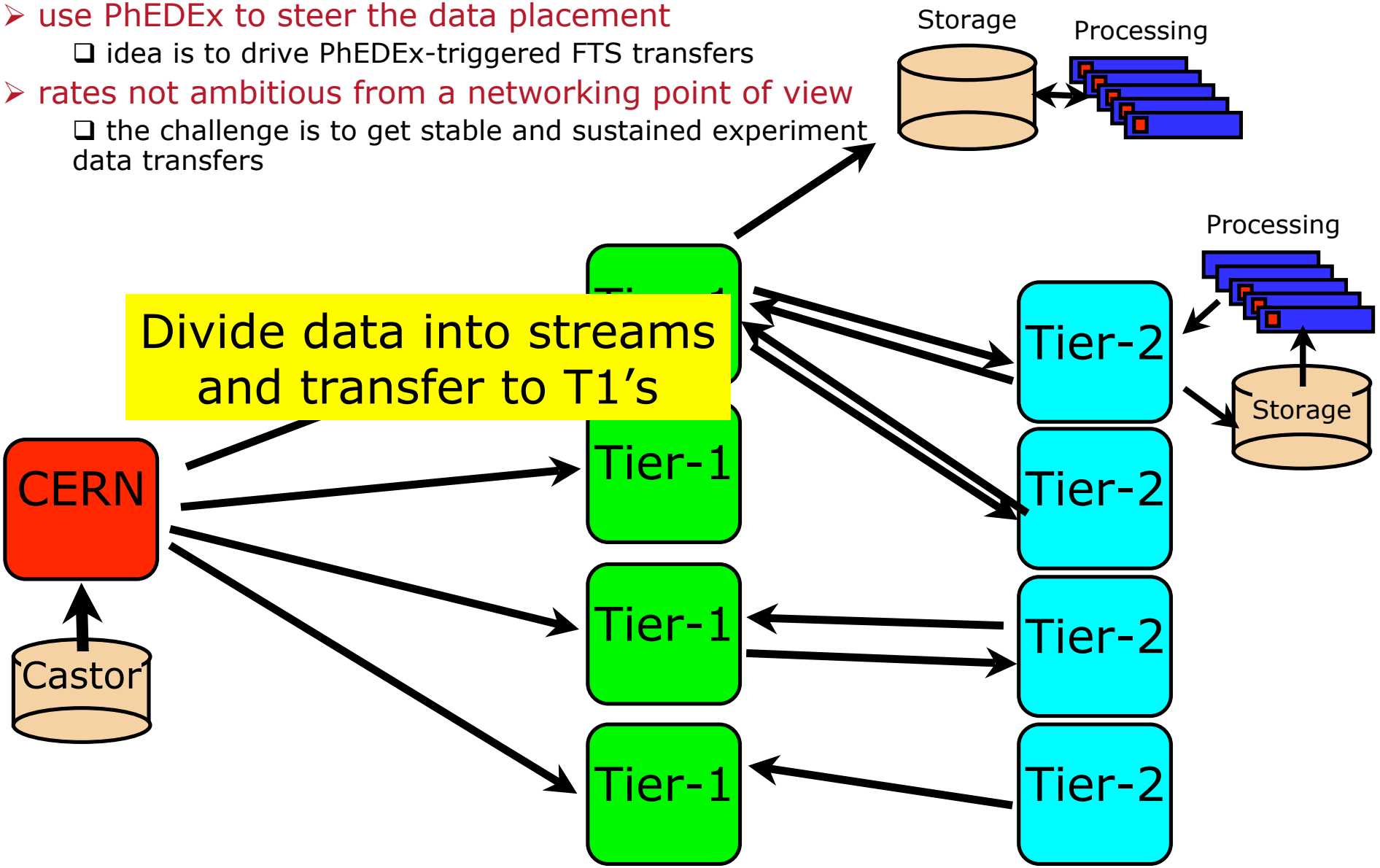
- Aims to demonstrate that all of the offline data processing requirements expressed in the exps' Computing Models, from raw data taking through to data access, can be handled within Grid at the full nominal data rate of the LHC
  - when: Apr 06 - Sep 06
  - who: T0, all T1's, majority of T2's

It will become the initial production service for LHC and made available to the exps for final testing, commissioning and processing of cosmic ray data

- Set-up ("throughput") phase (Apr 06)
  - Throughput sustaining for 3 weeks the target data rates at each site
    - Target is a stable, reliable data transfer to T1's at target rates to any supported SRM implementation (dCache, Castor, …) + factor 2 for backlogs/peaks.
- Service phase (May-Sep 06)
  - get the basic sw components required for the initial LHC data processing service into the loop
    - Target is to show capability to support full Computing Models of each LHC exp, from simulation to end-user batch analysis at Tier-2's

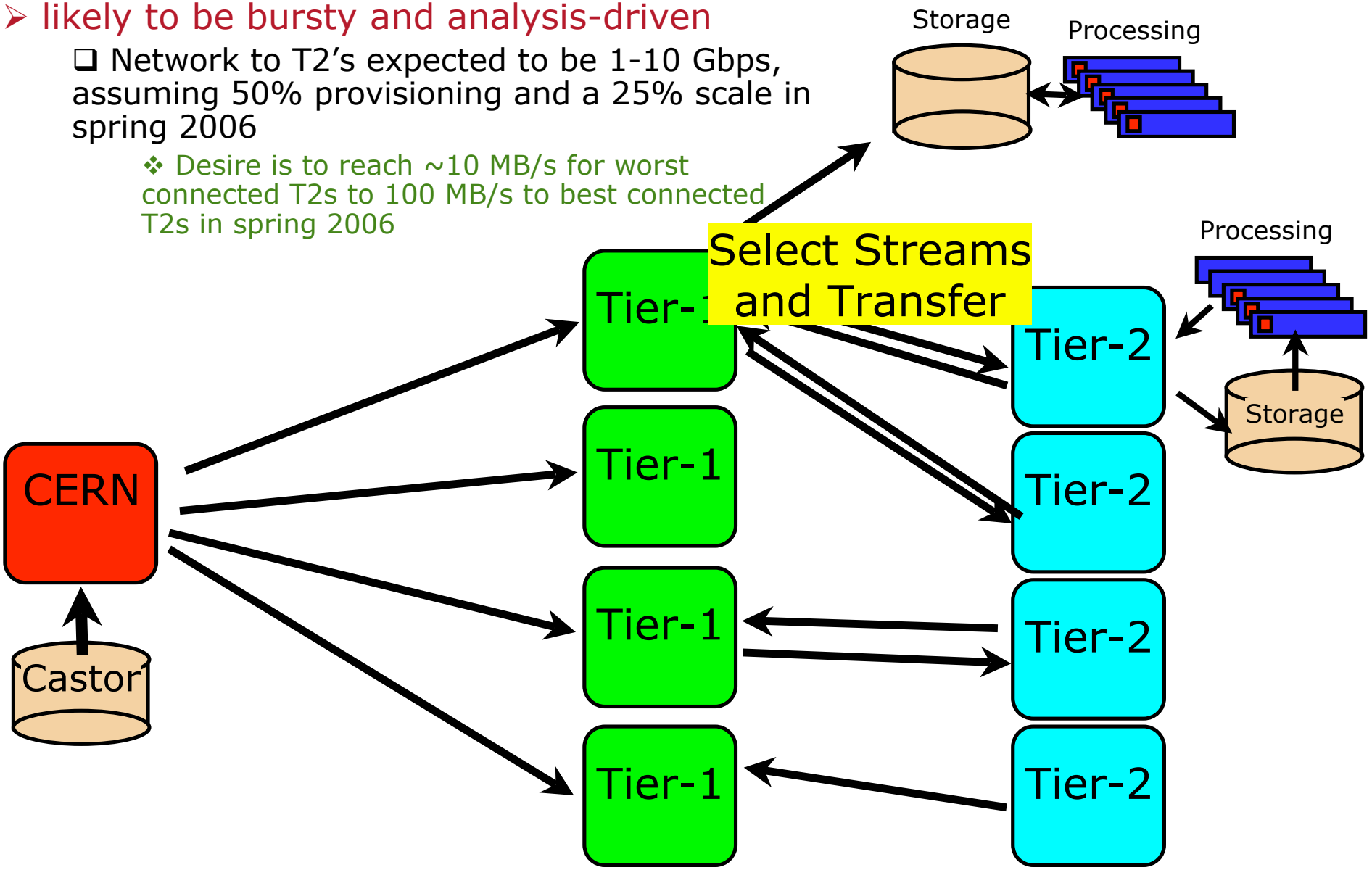
CMS: T0→T1 flows

- use PhEDEx to steer the data placement
  - idea is to drive PhEDEx-triggered FTS transfers
- rates not ambitious from a networking point of view
  - the challenge is to get stable and sustained experiment data transfers



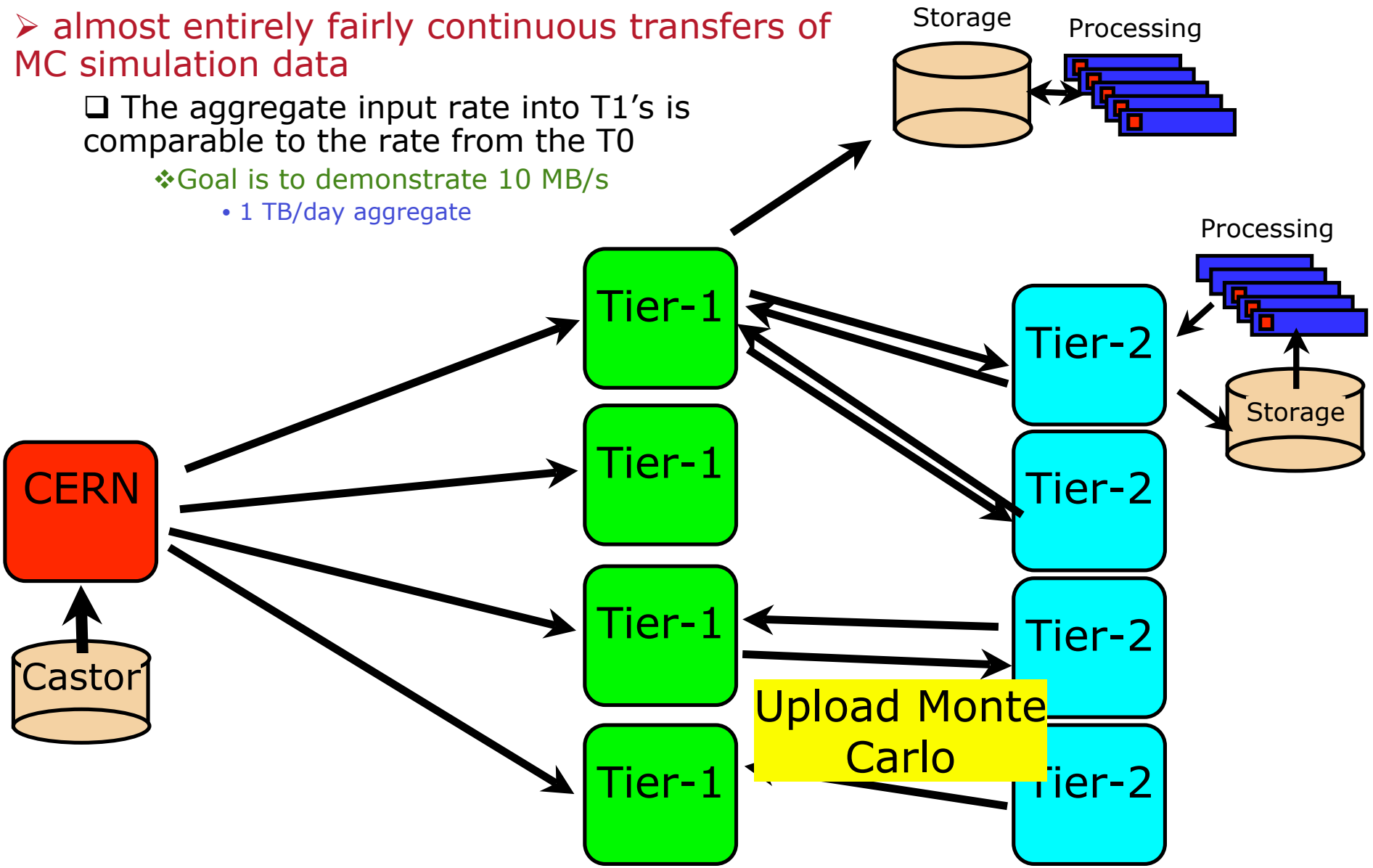
CMS: T1's→T2's flows

- likely to be bursty and analysis-driven
  - Network to T2's expected to be 1-10 Gbps, assuming 50% provisioning and a 25% scale in spring 2006
    - Desire is to reach ~10 MB/s for worst connected T2s to 100 MB/s to best connected T2s in spring 2006



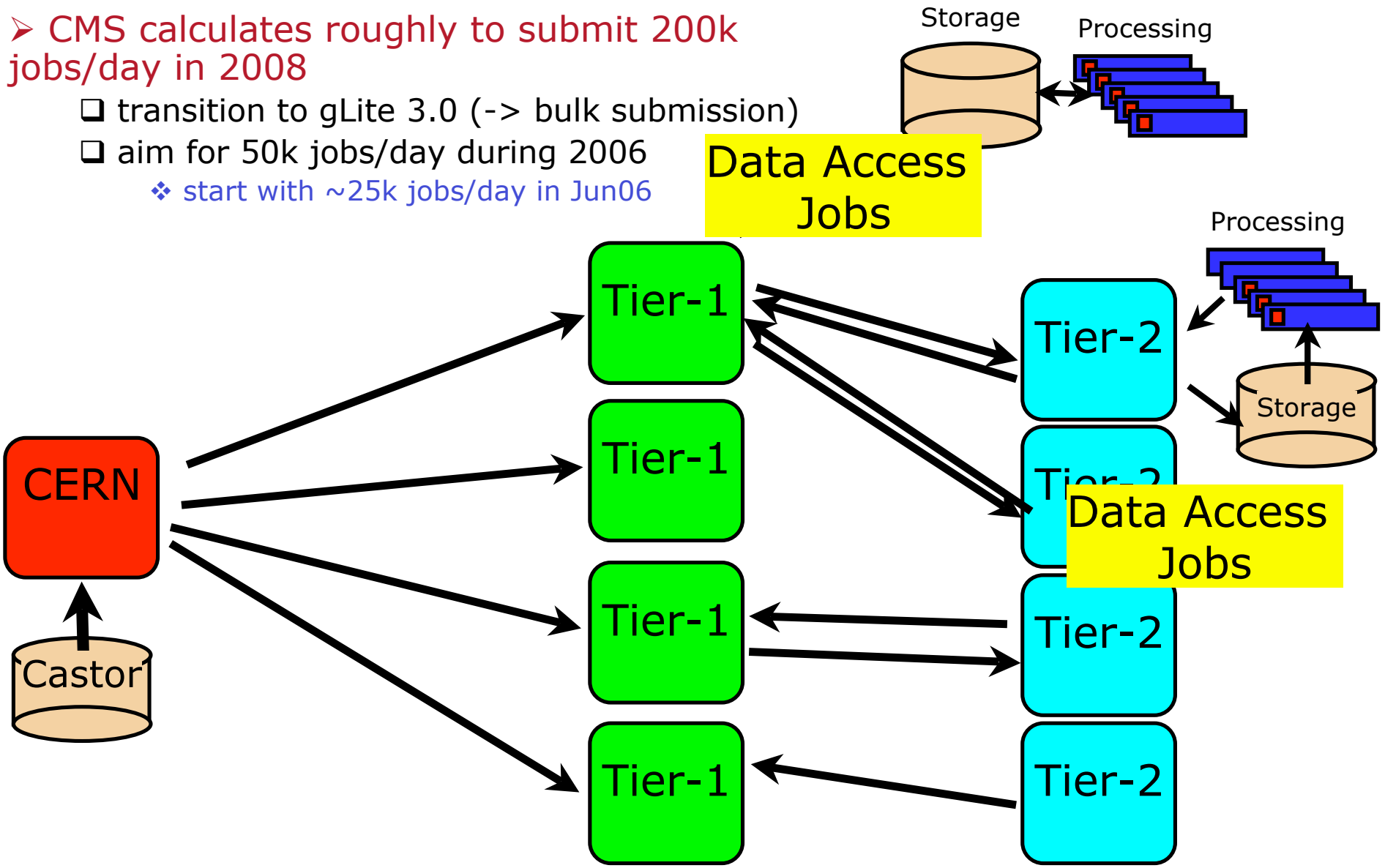
CMS: T2's→T1's flows

- almost entirely fairly continuous transfers of MC simulation data
  - The aggregate input rate into T1's is comparable to the rate from the T0
    - Goal is to demonstrate 10 MB/s
      - 1 TB/day aggregate



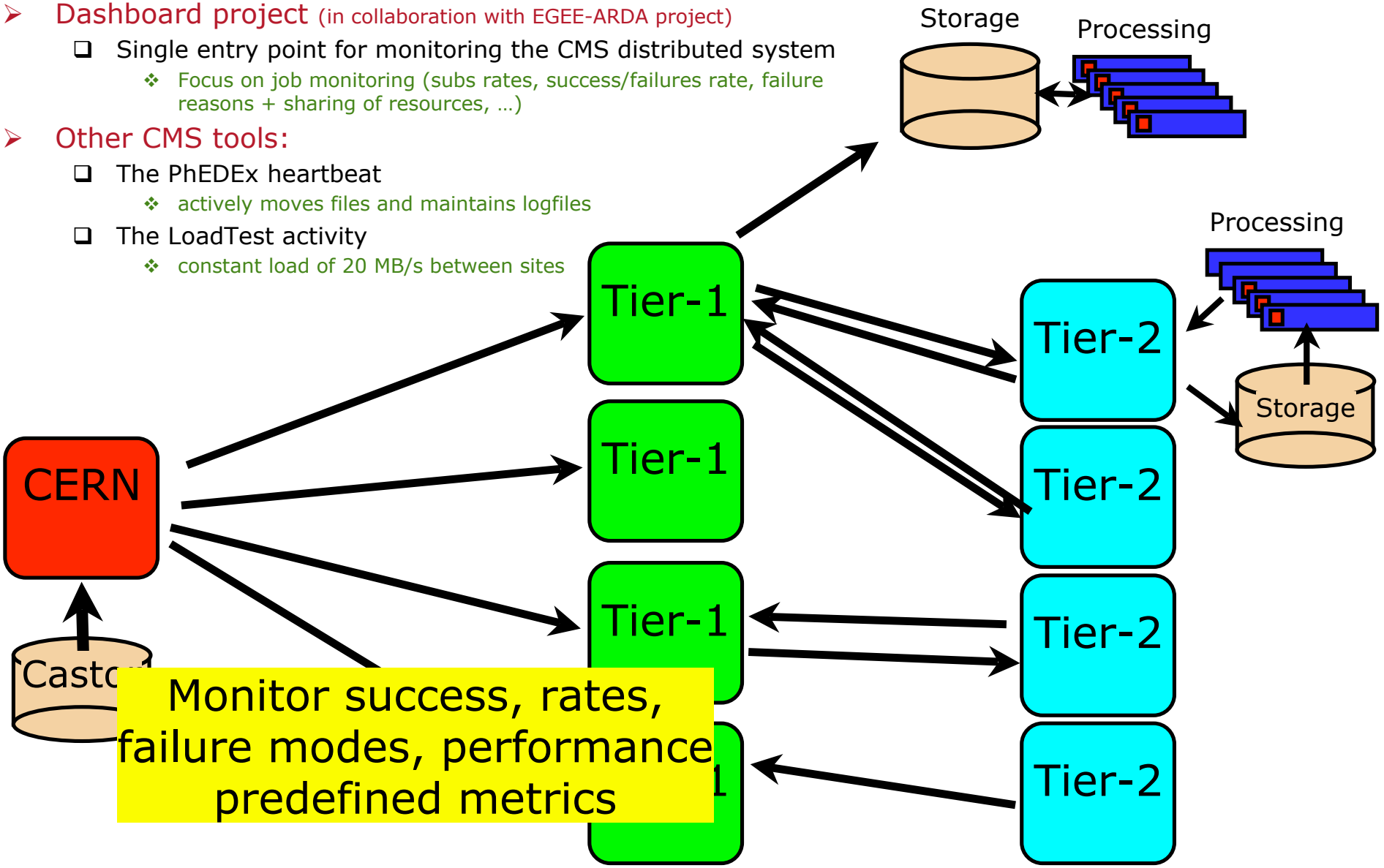
CMS: accessing the data

- CMS calculates roughly to submit 200k jobs/day in 2008
  - transition to gLite 3.0 (-> bulk submission)
  - aim for 50k jobs/day during 2006
    - start with ~25k jobs/day in Jun06

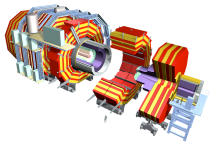


CMS: processes monitoring

- Dashboard project (in collaboration with EGEE-ARDA project)
  - Single entry point for monitoring the CMS distributed system
    - Focus on job monitoring (subs rates, success/failures rate, failure reasons + sharing of resources, …)
- Other CMS tools:
  - The PhEDEx heartbeat
    - actively moves files and maintains logfiles
  - The LoadTest activity
    - constant load of 20 MB/s between sites



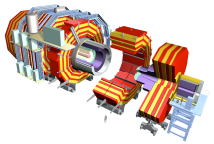
Computing, Software & Analysis Challenge 2006



- CSA06 is designed to demonstrate the computing system at a scale of 25% that of 2008
  - Computing Systems commissioning
  - Validation new data processing FW, new EDM …
- More a "capacity" challenge than a "complexity" challenge
  - parameters under definitions right now
  - CMS participates to SC4 as a step towards CSA06
    - Not a pass-or-fail test, but - as before - another step of an iterative activity to spawn the areas of work and drive the processes
- (rough) CMS schedule in 2006:

Apr	throughput phase for disk-to-disk and disk-to-tape transfers roll-out of new framework/EDM, new DMS, new MC production system
May	roll-out of gLite 3.0 open MC production system for debugging (LCG/OSG/local) 10 Mevts (usable evts) delivered (primary goal is validation of sw/EDM)
Jun	1st half: full CMS computing model functionality test in SC4 2nd half: large-scale production test for CMS (+ tail from 1st half)
Jul-Aug	50 Mevts delivered for CSA06 partial re-run of some SC3 activities (and tails of Jun06 SC4)
> Sep	CSA06 preparation and start

Summary



- CMS has adopted a distributed computing model making use of Grid technologies
- Steadily increase in scale and complexity
- Major changes in computing systems being done
  - DMS, processing framework/EDM, MC production system, …
- Major computing challenges ahead (SC4, CSA06)